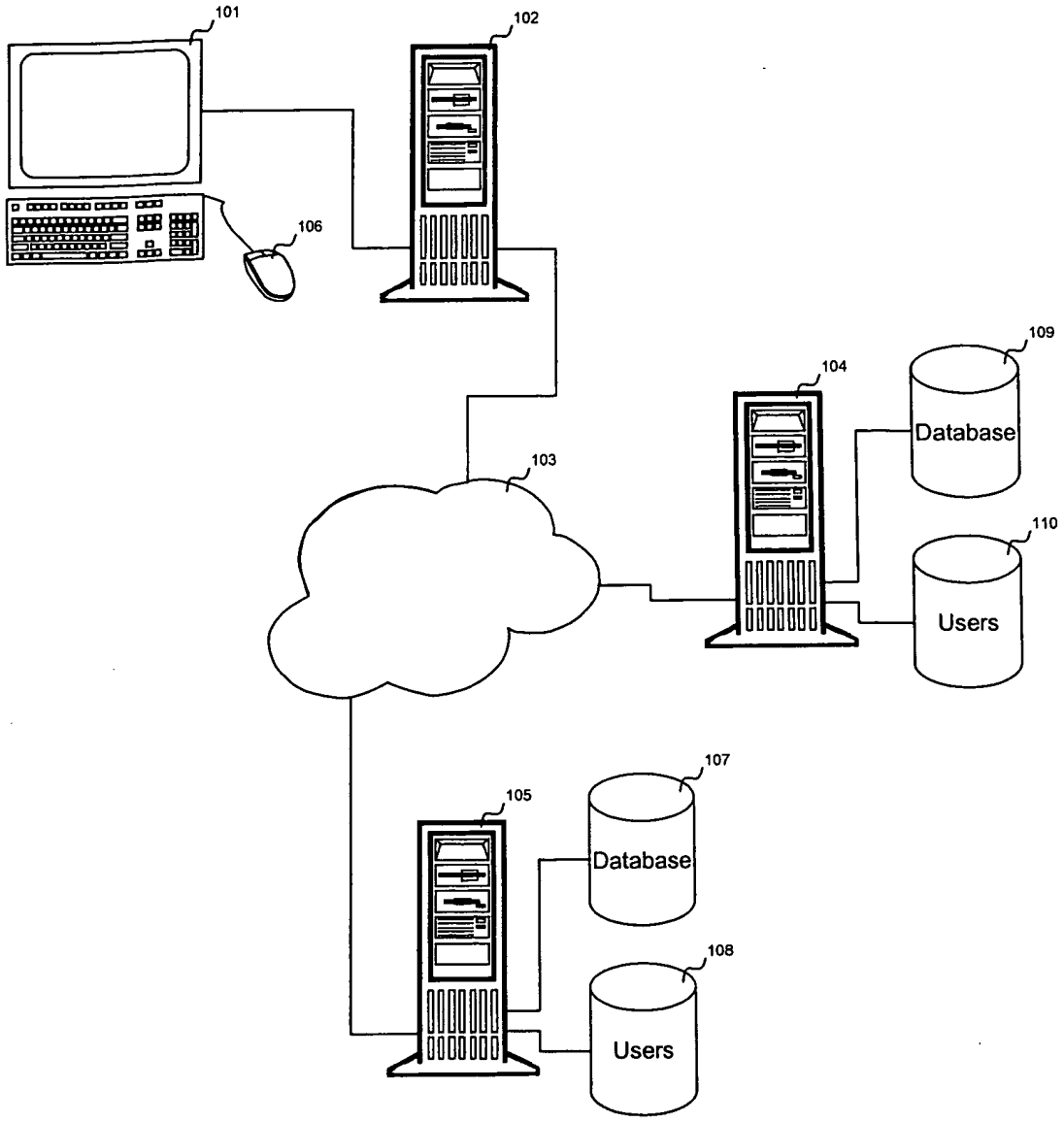

101
102
106
109
104
Database
103
110
Users
107
105
Database
108
Users

Figure 1

```
int TRAILER_calculate_trailer(int ct_x[],
                              int ct_y[],
                              int ct_t[],
                              int ct_length,
                              int *ct_a,
                              int *ct_b)
{ int i;

  int ct_idx1=TRIALER_INIT_INDEX1,ct_idx2=TRIALER_INIT_INDEX2,
      ct_idx3=TRIALER_INIT_INDEX3,ct_idx4=TRIALER_INIT_INDEX4;

  int ct_idx5=TRIALER_INIT_INDEX5,ct_idx6= TRIALER_INIT_INDEX6,
      ct_idx7=TRIALER_INIT_INDEX7,ct_idx8=TRIALER_INIT_INDEX8;

  int ct_sum1=0,ct_sum2=0,ct_sum3=0,ct_sum4=0;

  int ct_prime[N]=
  { TRAILER_PRIME_11, TRAILER_PRIME_12, ..., TRAILER_PRIME_1N
  };

  int ct_prime1[N1]=
  { TRAILER_PRIME_21, TRAILER_PRIME_22, ..., TRAILER_PRIME_2N1
  };

  if (!ct_x || !ct_y || !ct_t ||
      ct_length<40 || !ct_a || !ct_b) return 0;

  for(i=0;i<ct_length;++i)
  {
    // Second and third order sums.
    ct_sum1+=ct_prime[ct_idx1]*ct_prime1[ct_idx5];
    ct_sum2+=ct_prime[ct_idx2]*ct_prime1[ct_idx6];
    ct_sum3+=ct_prime[ct_idx3]*ct_prime1[ct_idx7];
    ct_sum4+=
    ct_prime[ct_idx6]* ct_prime[ct_idx4]*ct_prime1[ct_idx8]*
ct_x[i]*ct_y[i]*ct_t[i];

  }

  return 1;
}
```

Figure 2

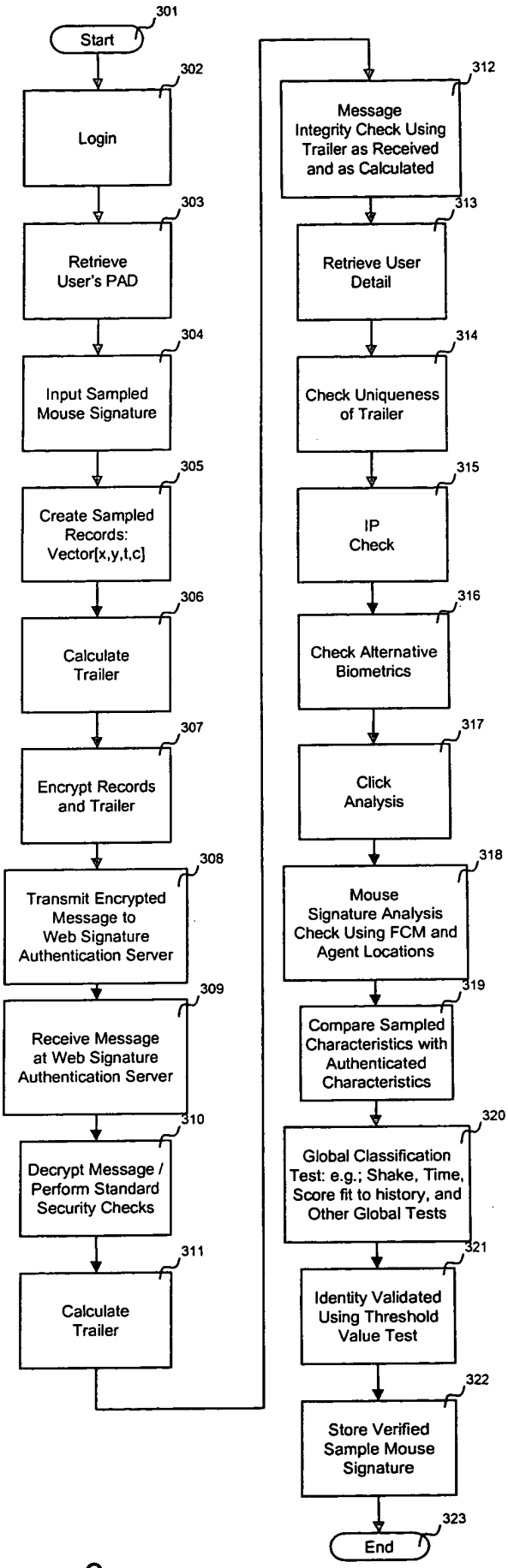

301
Start
302
Login
303
Retrieve User's PAD
304
Input Sampled Mouse Signature
305
Create Sampled Records: Vector[x,y,t,c]
306
Calculate Trailer
307
Encrypt Records and Trailer
308
Transmit Encrypted Message to Web Signature Authentication Server
309
Receive Message at Web Signature Authentication Server
310
Decrypt Message / Perform Standard Security Checks
311
Calculate Trailer
312
Message Integrity Check Using Trailer as Received and as Calculated
313
Retrieve User Detail
314
Check Uniqueness of Trailer
315
IP Check
316
Check Alternative Biometrics
317
Click Analysis
318
Mouse Signature Analysis Check Using FCM and Agent Locations
319
Compare Sampled Characteristics with Authenticated Characteristics
320
Global Classification Test: e.g.; Shake, Time, Score fit to history, and Other Global Tests
321
Identity Validated Using Threshold Value Test
322
Store Verified Sample Mouse Signature
323
End

Figure 3

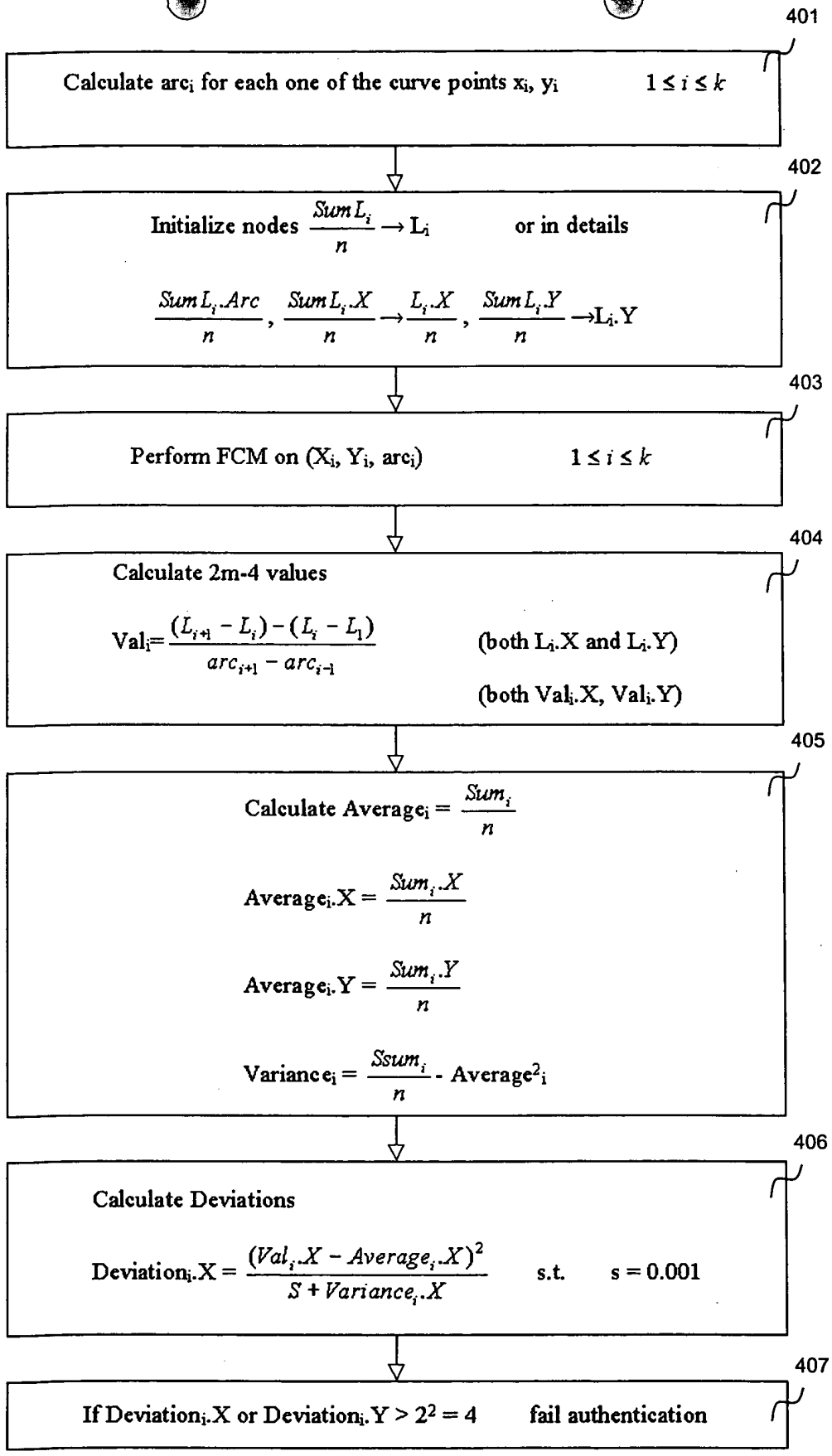

401
Calculate arc$_i$ for each one of the curve points x_i, y_i $1 \le i \le k$
402
Initialize nodes $\frac{SumL_i}{n} \rightarrow L_i$ or in details
$\frac{SumL_i.Arc}{n}, \frac{SumL_i.X}{n} \rightarrow \frac{L_i.X}{n}, \frac{SumL_i.Y}{n} \rightarrow L_i.Y$
403
Perform FCM on (X_i, Y_i, arc$_i$) $1 \le i \le k$
404
Calculate 2m-4 values
$Val_i = \frac{(L_{i+1} - L_i) - (L_i - L_1)}{arc_{i+1} - arc_{i-1}}$ (both $L_i.X$ and $L_i.Y$)
(both $Val_i.X$, $Val_i.Y$)
405
Calculate Average$_i$ = $\frac{Sum_i}{n}$
Average$_i$.X = $\frac{Sum_i.X}{n}$
Average$_i$.Y = $\frac{Sum_i.Y}{n}$
Variance$_i$ = $\frac{Ssum_i}{n}$ - Average^2_i
406
Calculate Deviations
Deviation$_i$.X = $\frac{(Val_i.X - Average_i.X)^2}{S + Variance_i.X}$ s.t. s = 0.001
407
If Deviation$_i$.X or Deviation$_i$.Y > $2^2 = 4$ fail authentication

Figure 4

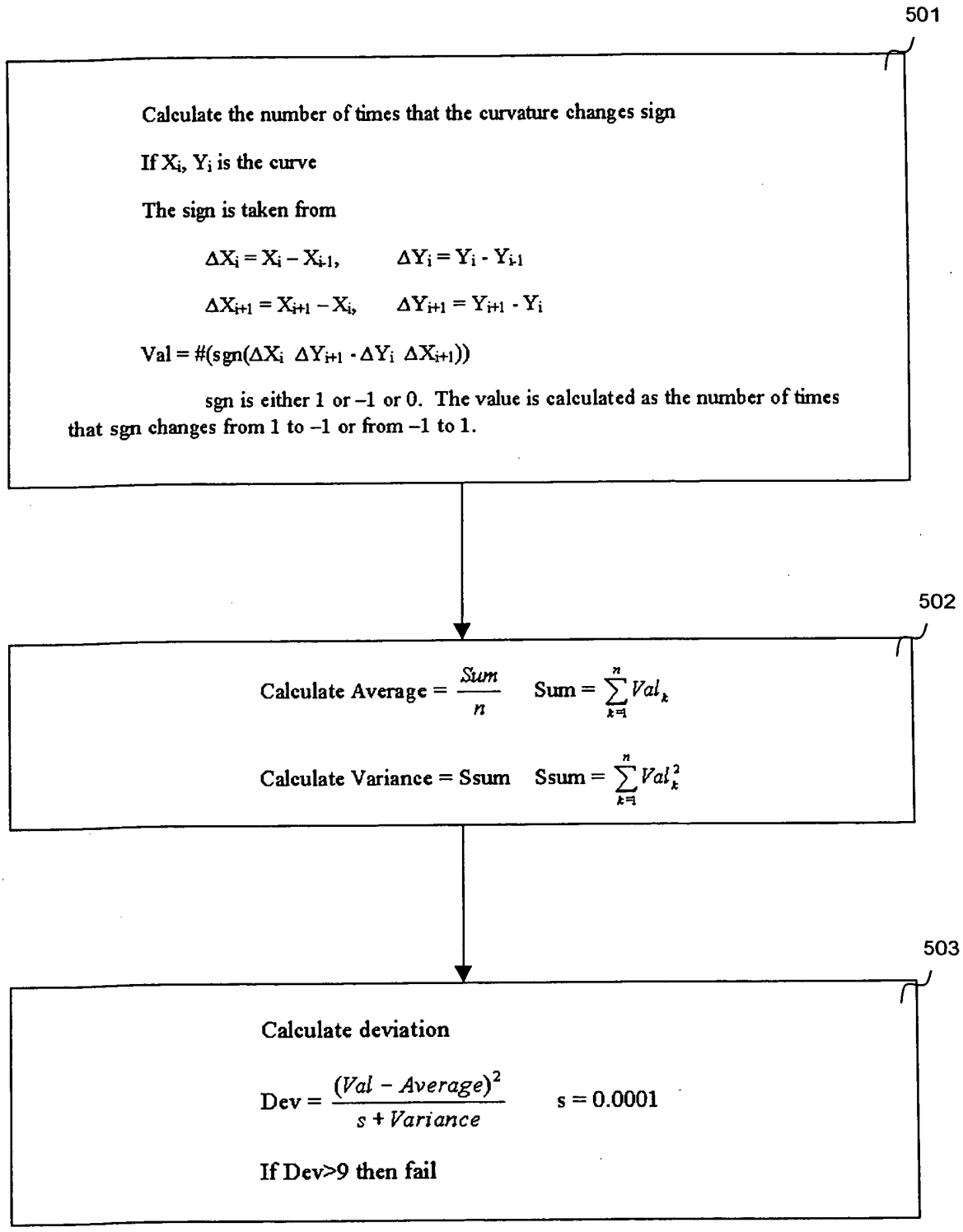

501
Calculate the number of times that the curvature changes sign
If X_i, Y_i is the curve
The sign is taken from
$\Delta X_i = X_i - X_{i-1}$, $\Delta Y_i = Y_i - Y_{i-1}$
$\Delta X_{i+1} = X_{i+1} - X_i$, $\Delta Y_{i+1} = Y_{i+1} - Y_i$
$Val = \#(sgn(\Delta X_i \Delta Y_{i+1} - \Delta Y_i \Delta X_{i+1}))$
sgn is either 1 or –1 or 0. The value is calculated as the number of times that sgn changes from 1 to –1 or from –1 to 1.
502
Calculate Average = $\frac{Sum}{n}$ Sum = $\sum_{k=1}^{n} Val_k$
Calculate Variance = Ssum Ssum = $\sum_{k=1}^{n} Val_k^2$
503
Calculate deviation
Dev = $\frac{(Val - Average)^2}{s + Variance}$ s = 0.0001
If Dev>9 then fail

Figure 5

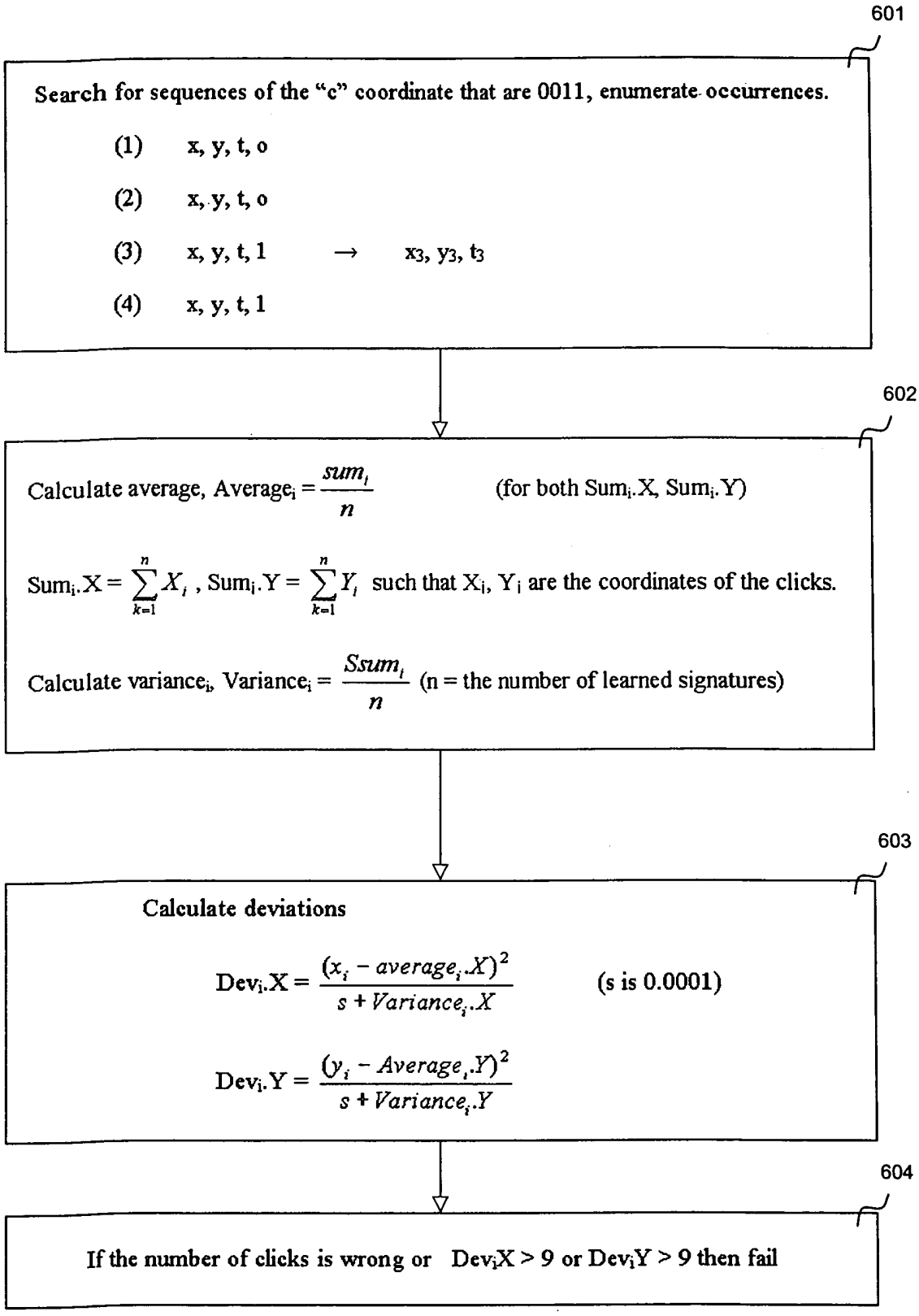

601
Search for sequences of the "c" coordinate that are 0011, enumerate occurrences.
(1) x, y, t, o
(2) x, y, t, o
(3) x, y, t, 1 → x3, y3, t3
(4) x, y, t, 1
602
Calculate average, Average_i = sum_i / n (for both Sum_i.X, Sum_i.Y)
Sum_i.X = Σ_{k=1}^{n} X_i , Sum_i.Y = Σ_{k=1}^{n} Y_i such that X_i, Y_i are the coordinates of the clicks.
Calculate variance_i, Variance_i = Ssum_i / n (n = the number of learned signatures)
603
Calculate deviations
Dev_i.X = (x_i − average_i.X)^2 / (s + Variance_i.X) (s is 0.0001)
Dev_i.Y = (y_i − Average_i.Y)^2 / (s + Variance_i.Y)
604
If the number of clicks is wrong or Dev_iX > 9 or Dev_iY > 9 then fail

Figure 6

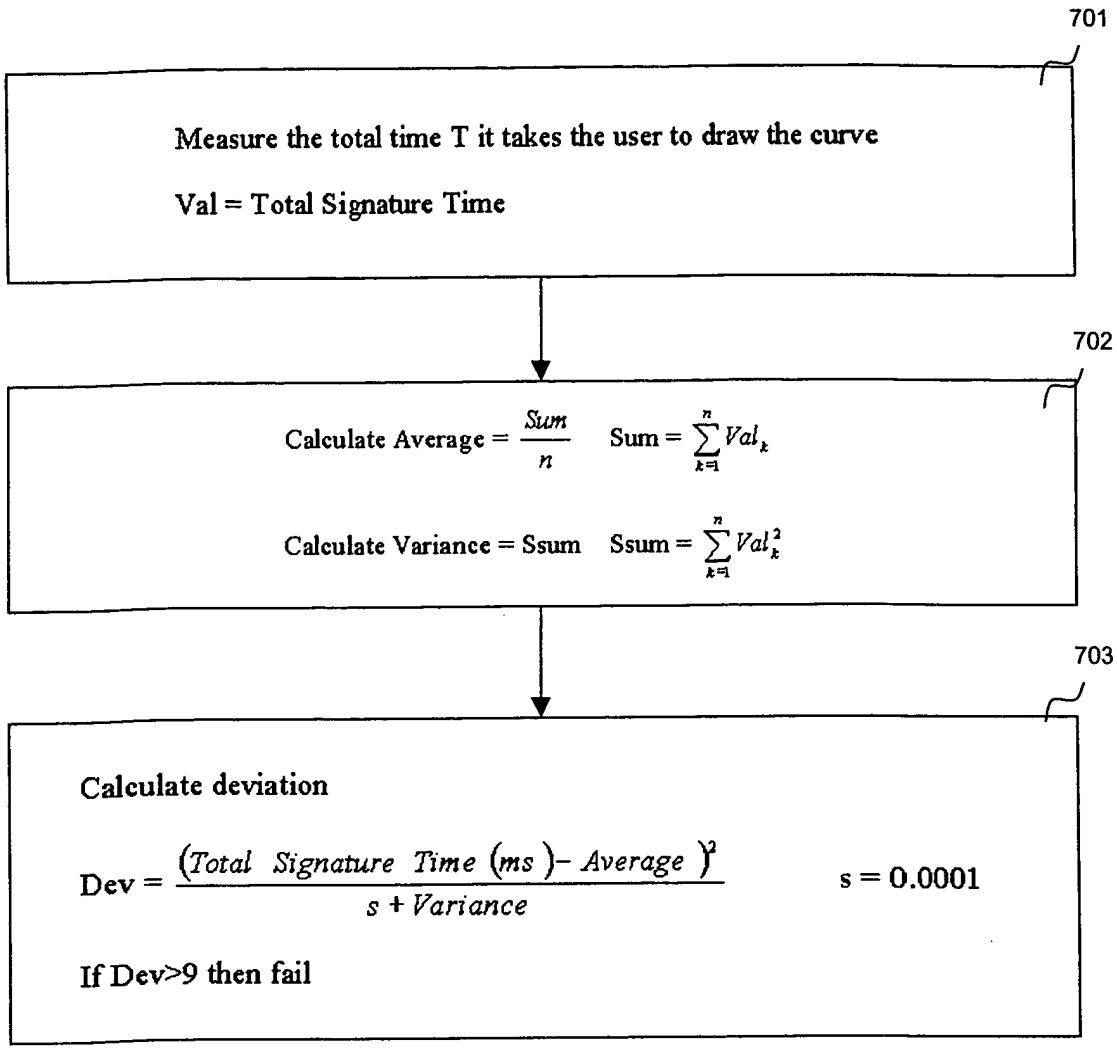

701
Measure the total time T it takes the user to draw the curve
Val = Total Signature Time
702
Calculate Average = $\frac{Sum}{n}$ Sum = $\sum_{k=1}^{n} Val_k$
Calculate Variance = Ssum Ssum = $\sum_{k=1}^{n} Val_k^2$
703
Calculate deviation
Dev = $\frac{(Total\ Signature\ Time\ (ms) - Average)^2}{s + Variance}$ s = 0.0001
If Dev>9 then fail

Figure 7

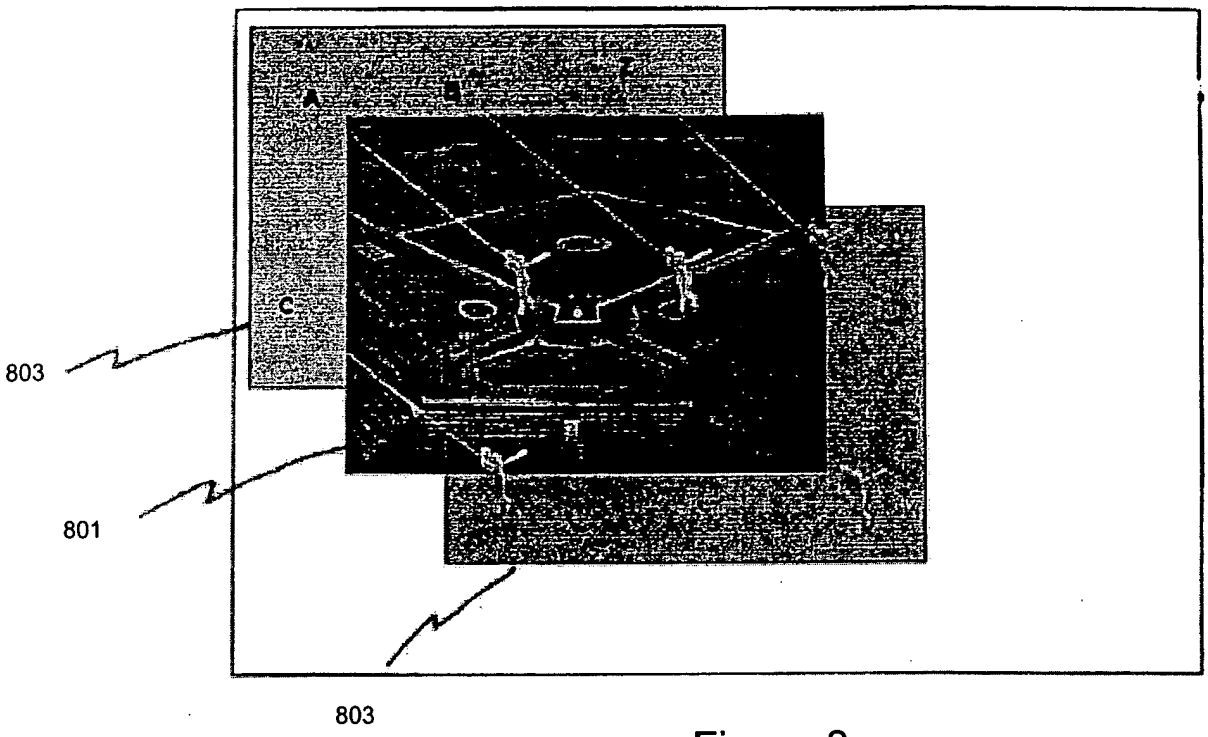


Figure 8

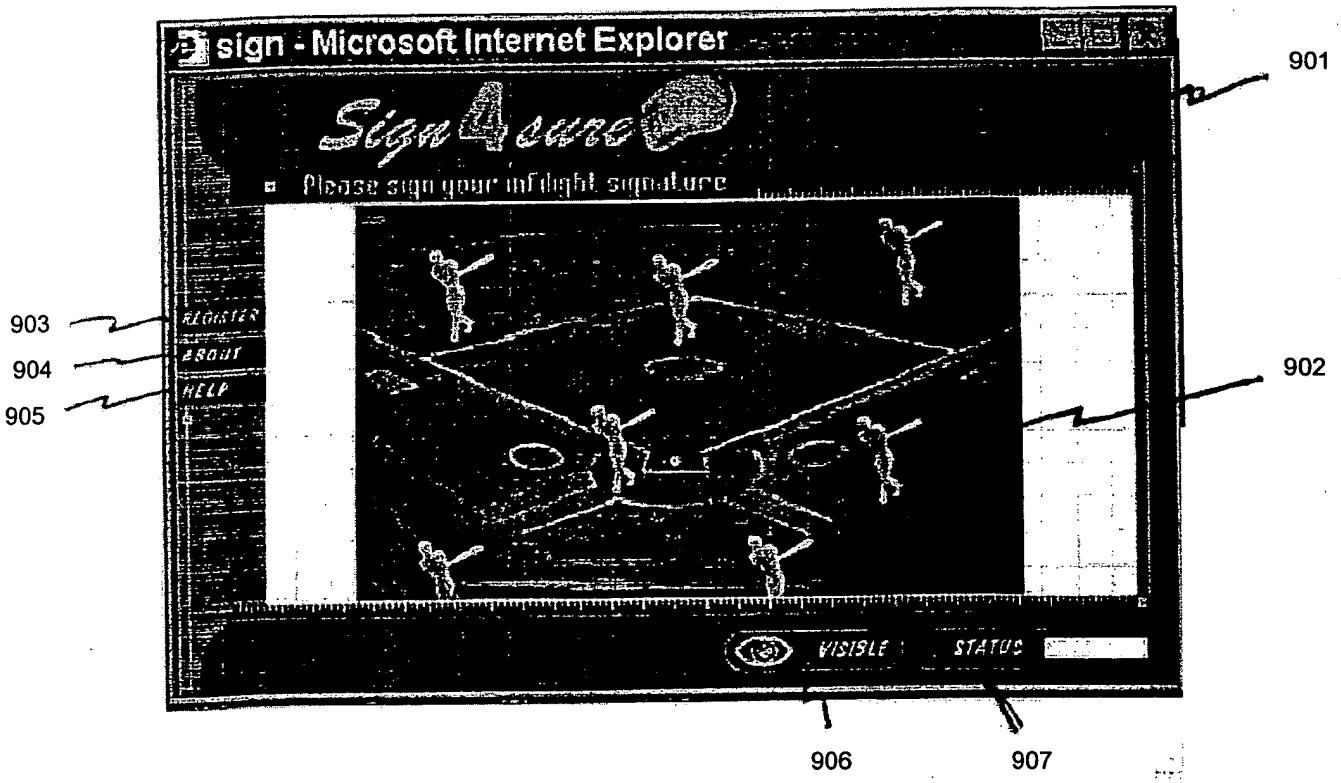


Figure 9

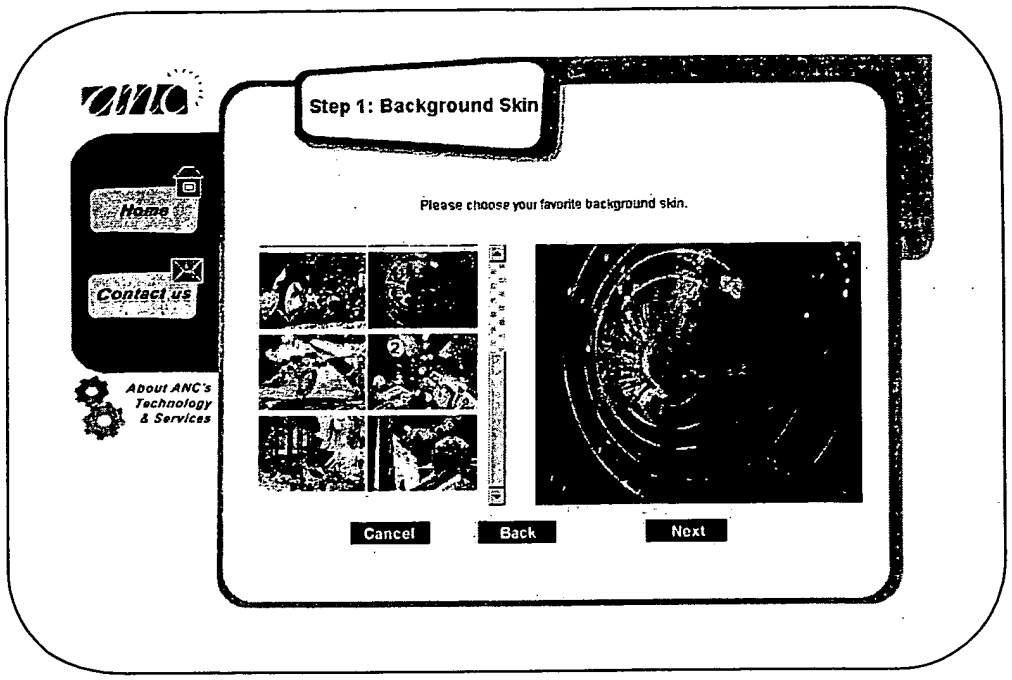


Figure 10

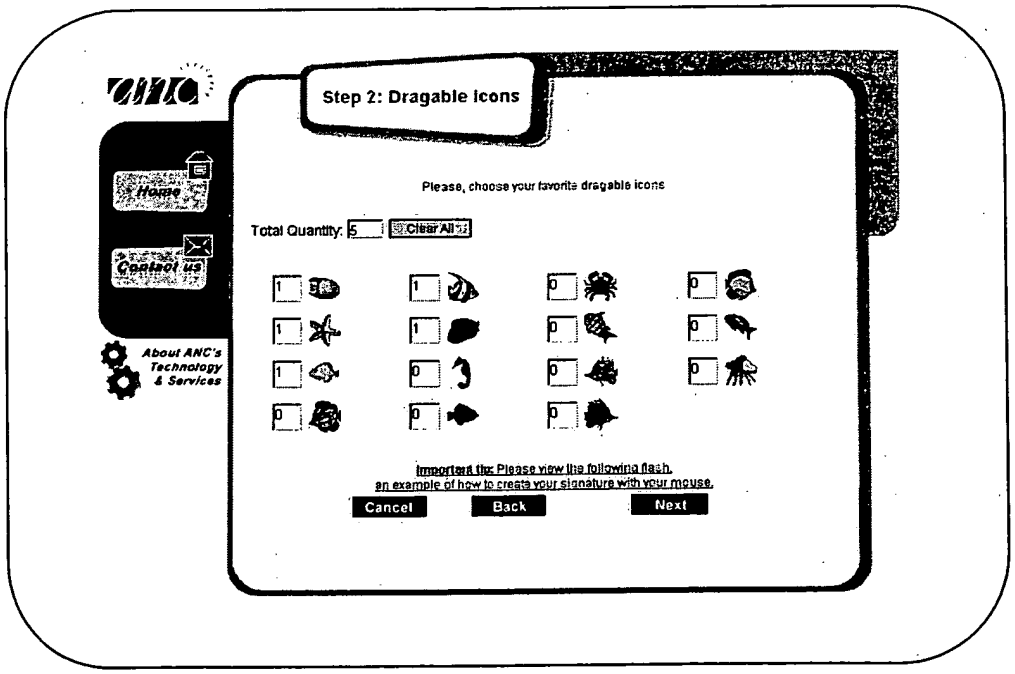


Figure 11

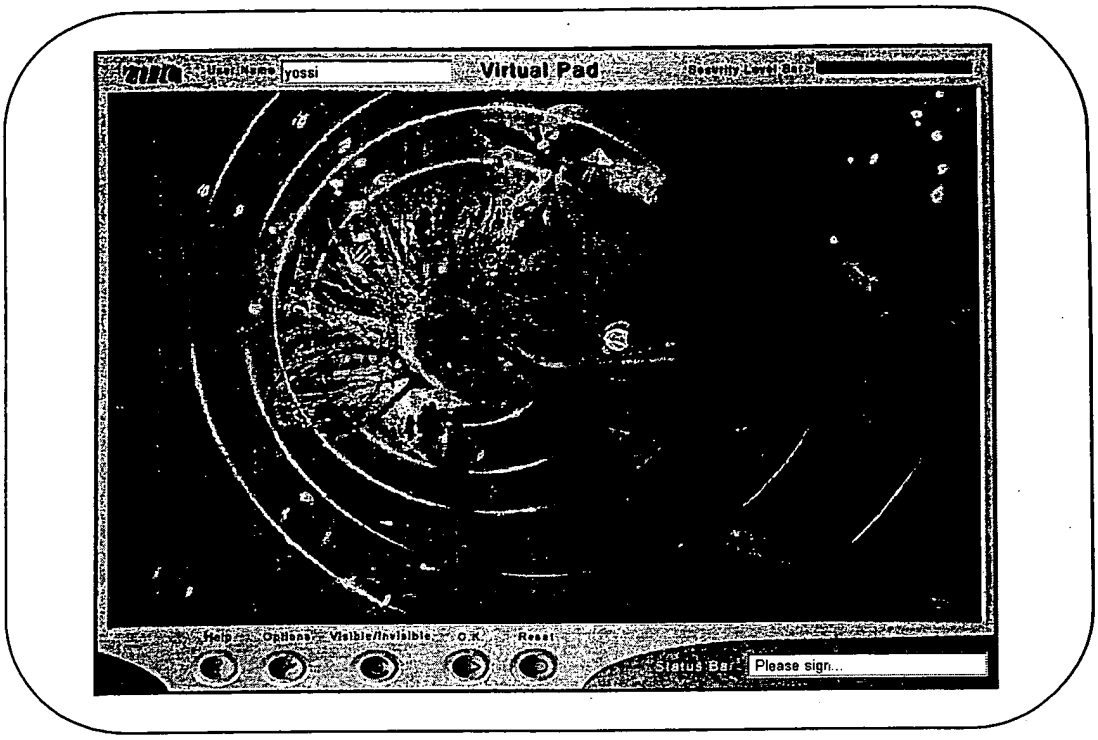


Figure 12

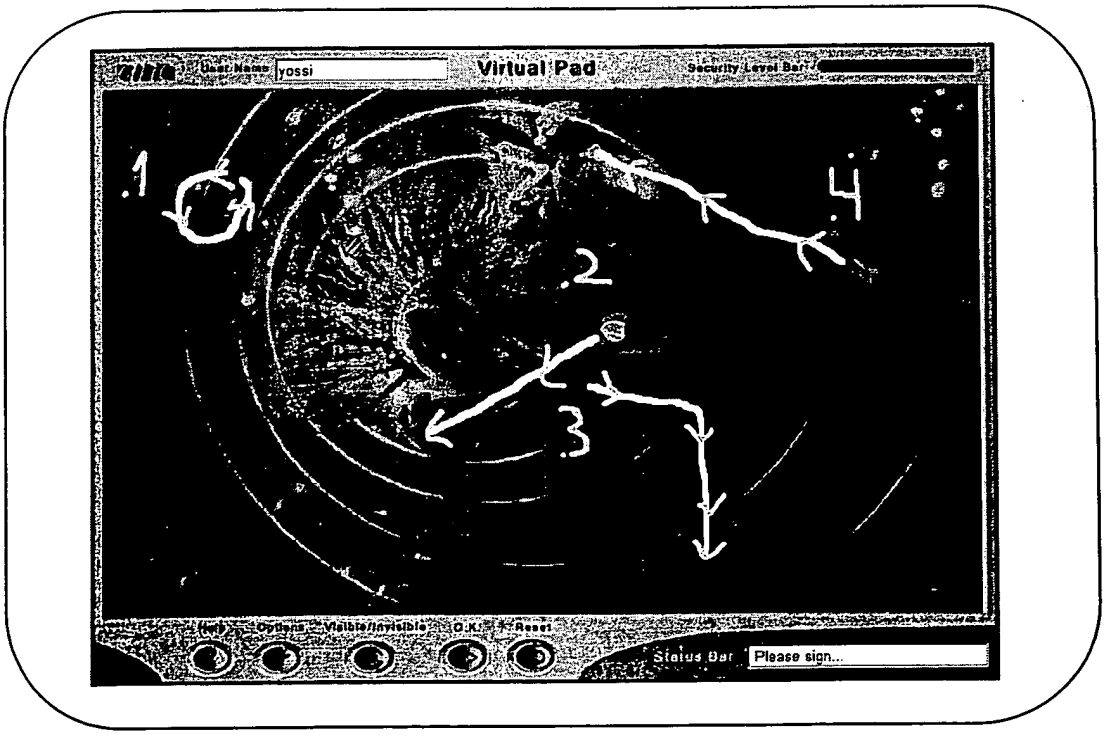


Figure 13

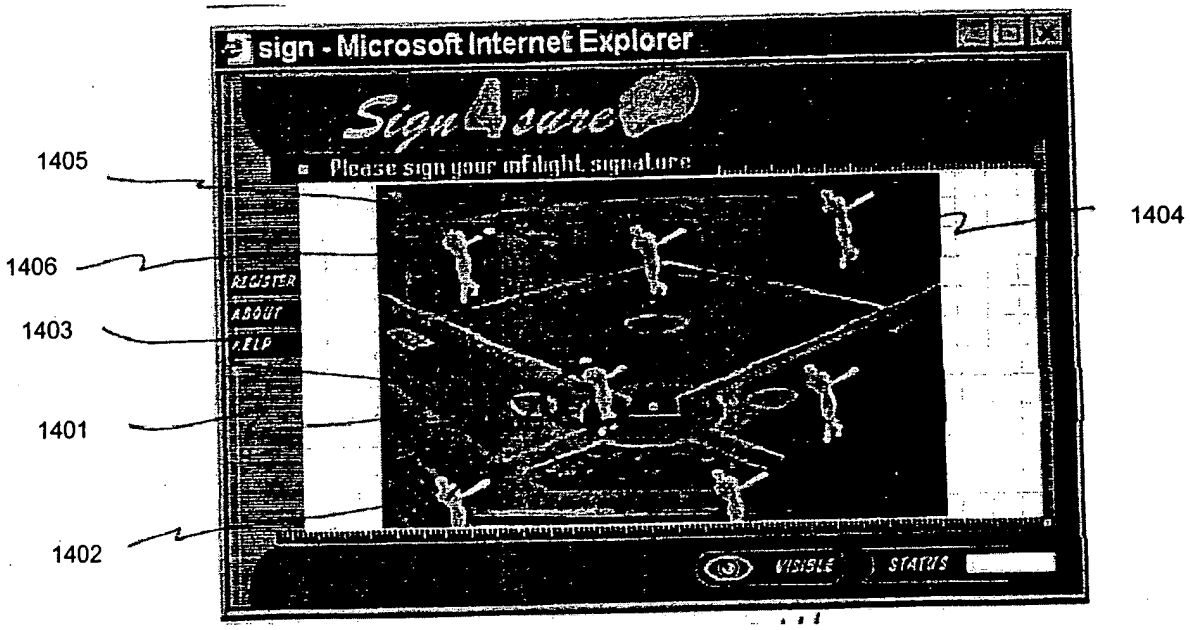

sign - Microsoft Internet Explorer
Sign4sure
Please sign your inflight signature
REGISTER
ABOUT
HELP
VISIBLE
STATUS
1405
1406
1403
1401
1402
1404

Figure14

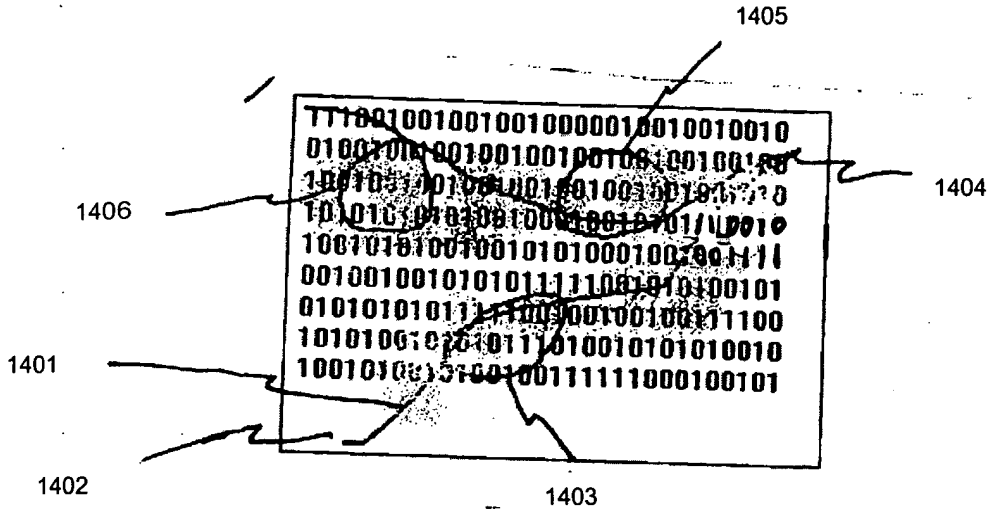


Figure 15

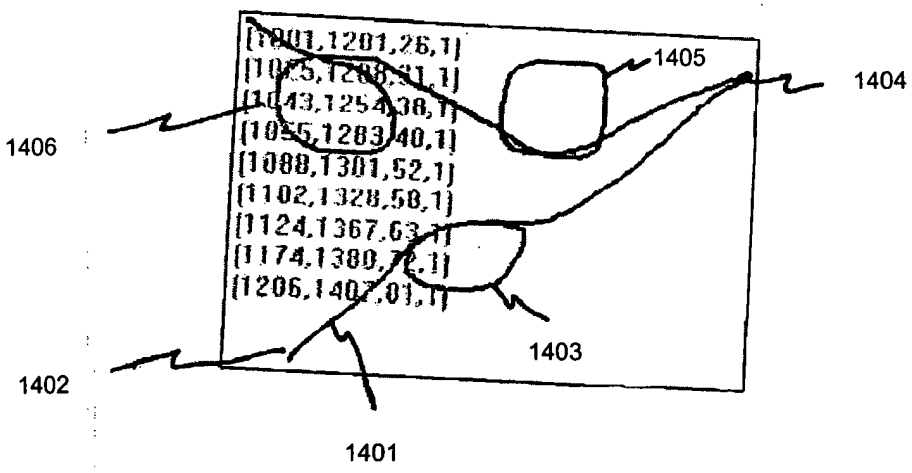


Figure 21

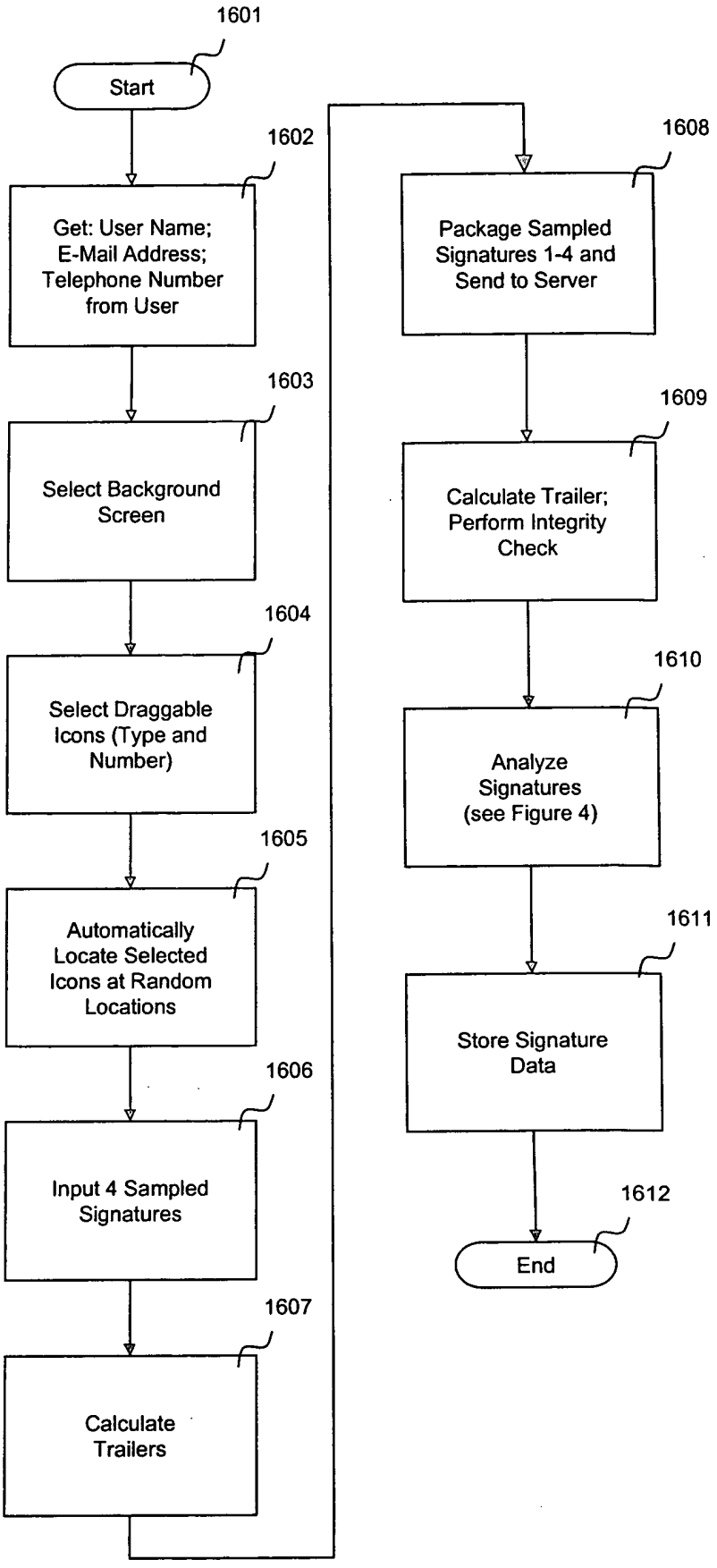

1601
Start
1602
Get: User Name; E-Mail Address; Telephone Number from User
1603
Select Background Screen
1604
Select Draggable Icons (Type and Number)
1605
Automatically Locate Selected Icons at Random Locations
1606
Input 4 Sampled Signatures
1607
Calculate Trailers
1608
Package Sampled Signatures 1-4 and Send to Server
1609
Calculate Trailer; Perform Integrity Check
1610
Analyze Signatures (see Figure 4)
1611
Store Signature Data
1612
End

Figure 16

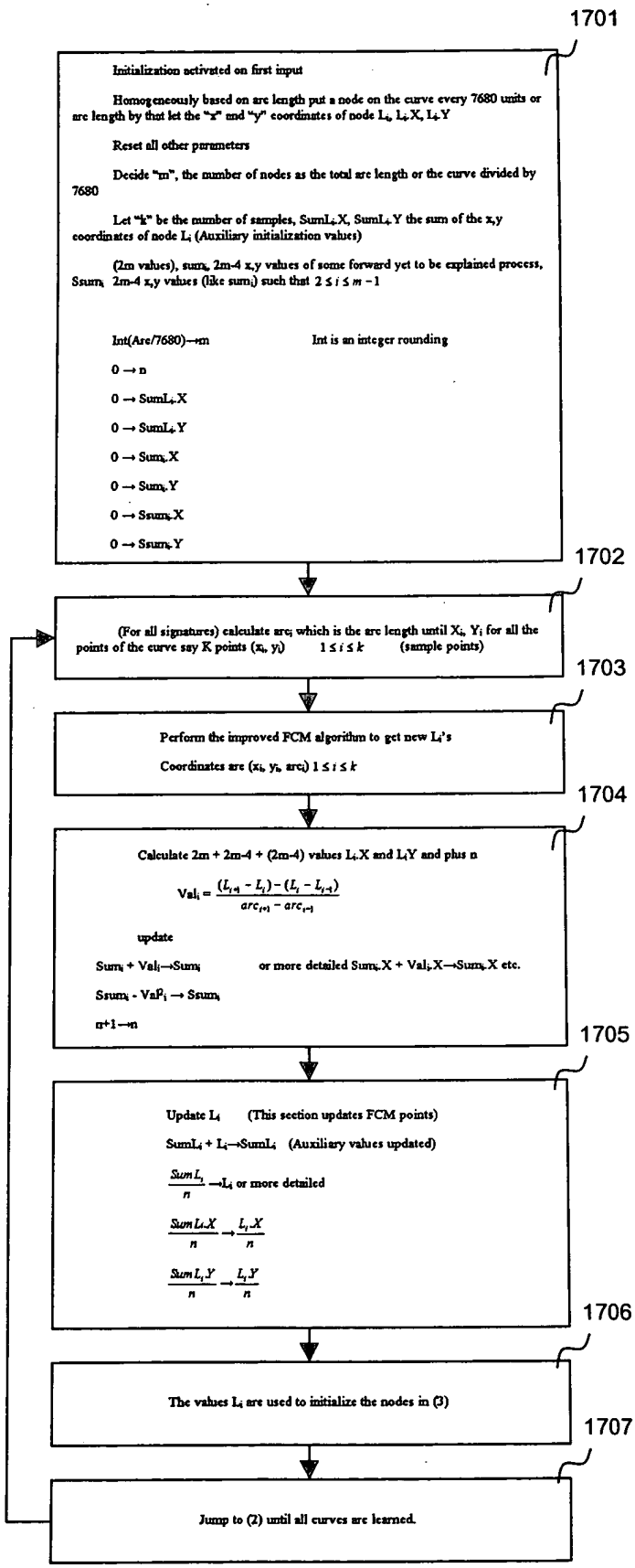

1701
Initialization activated on first input
Homogeneously based on arc length put a node on the curve every 7680 units or arc length by that let the "x" and "y" coordinates of node L_0, $L_0.X$, $L_0.Y$
Reset all other parameters
Decide "m", the number of nodes as the total arc length or the curve divided by 7680
Let "k" be the number of samples, $SumL_0.X$, $SumL_0.Y$ the sum of the x,y coordinates of node L_i (Auxiliary initialization values)
(2m values), sum_0 2m-4 x,y values of some forward yet to be explained process, $Ssum_0$ 2m-4 x,y values (like sum_0) such that $2 \le i \le m-1$
Int(Are/7680)→m Int is an integer rounding
0 → n
0 → $SumL_0.X$
0 → $SumL_0.Y$
0 → $Sum_0.X$
0 → $Sum_0.Y$
0 → $Ssum_0.X$
0 → $Ssum_0.Y$
1702
(For all signatures) calculate arc_i which is the arc length until X_i, Y_i for all the points of the curve say K points (x_i, y_i) $1 \le i \le k$ (sample points)
1703
Perform the improved FCM algorithm to get new L_i's
Coordinates are (x_i, y_i, arc_i) $1 \le i \le k$
1704
Calculate 2m + 2m-4 + (2m-4) values $L_i.X$ and $L_i.Y$ and plus n
$Val_i = \frac{(L_{i+1} - L_i) - (L_i - L_{i-1})}{arc_{i+1} - arc_{i-1}}$
update
$Sum_i + Val_i \rightarrow Sum_i$ or more detailed $Sum_i.X + Val_i.X \rightarrow Sum_i.X$ etc.
$Ssum_i - Val_i \rightarrow Ssum_i$
n+1→n
1705
Update L_i (This section updates FCM points)
$SumL_i + L_i \rightarrow SumL_i$ (Auxiliary values updated)
$\frac{SumL_i}{n} \rightarrow L_i$ or more detailed
$\frac{SumL_i.X}{n} \rightarrow \frac{L_i.X}{n}$
$\frac{SumL_i.Y}{n} \rightarrow \frac{L_i.Y}{n}$
1706
The values L_i are used to initialize the nodes in (3)
1707
Jump to (2) until all curves are learned.

Figure 17

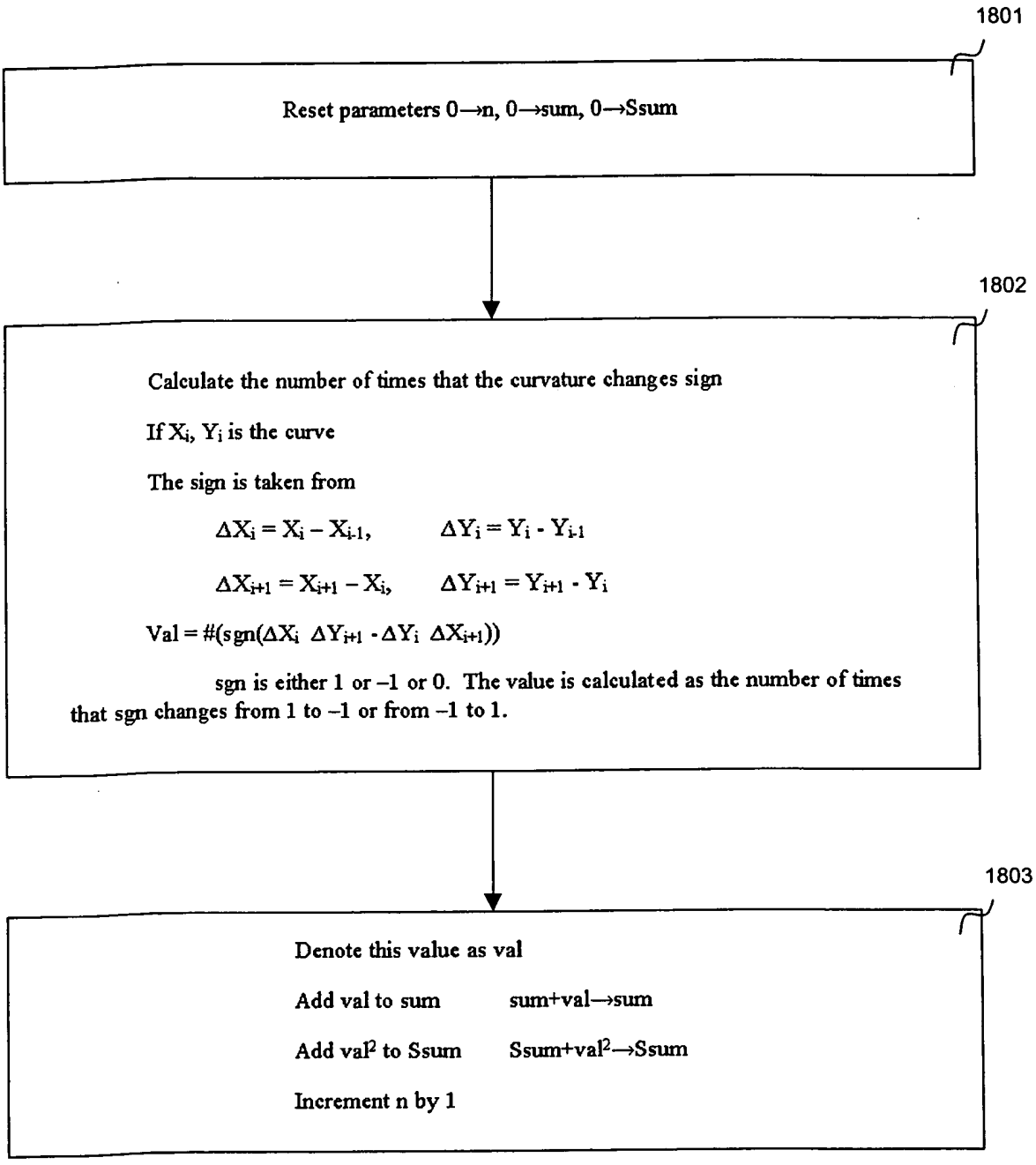

1801
Reset parameters 0→n, 0→sum, 0→Ssum
1802
Calculate the number of times that the curvature changes sign
If X_i, Y_i is the curve
The sign is taken from
$\Delta X_i = X_i - X_{i-1}$, $\Delta Y_i = Y_i - Y_{i-1}$
$\Delta X_{i+1} = X_{i+1} - X_i$, $\Delta Y_{i+1} = Y_{i+1} - Y_i$
$Val = \#(sgn(\Delta X_i \ \Delta Y_{i+1} - \Delta Y_i \ \Delta X_{i+1}))$
sgn is either 1 or −1 or 0. The value is calculated as the number of times that sgn changes from 1 to −1 or from −1 to 1.
1803
Denote this value as val
Add val to sum sum+val→sum
Add val^2 to Ssum $Ssum+val^2 \rightarrow Ssum$
Increment n by 1

Figure 18

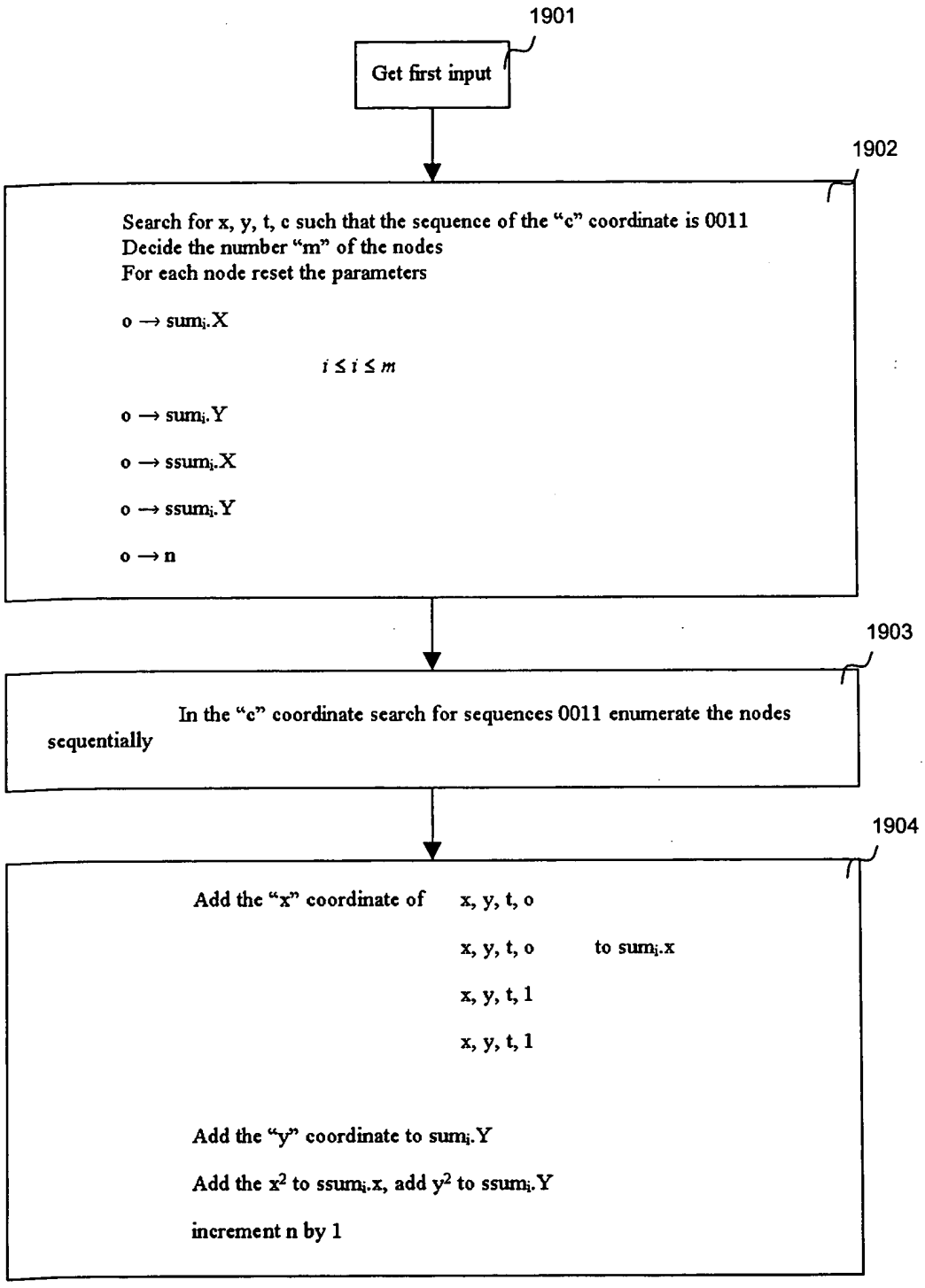

1901
Get first input
1902
Search for x, y, t, c such that the sequence of the "c" coordinate is 0011
Decide the number "m" of the nodes
For each node reset the parameters
o → sum$_i$.X
$i \le i \le m$
o → sum$_i$.Y
o → ssum$_i$.X
o → ssum$_i$.Y
o → n
1903
In the "c" coordinate search for sequences 0011 enumerate the nodes sequentially
1904
Add the "x" coordinate of x, y, t, o
x, y, t, o to sum$_i$.x
x, y, t, 1
x, y, t, 1
Add the "y" coordinate to sum$_i$.Y
Add the x^2 to ssum$_i$.x, add y^2 to ssum$_i$.Y
increment n by 1

Figure 19

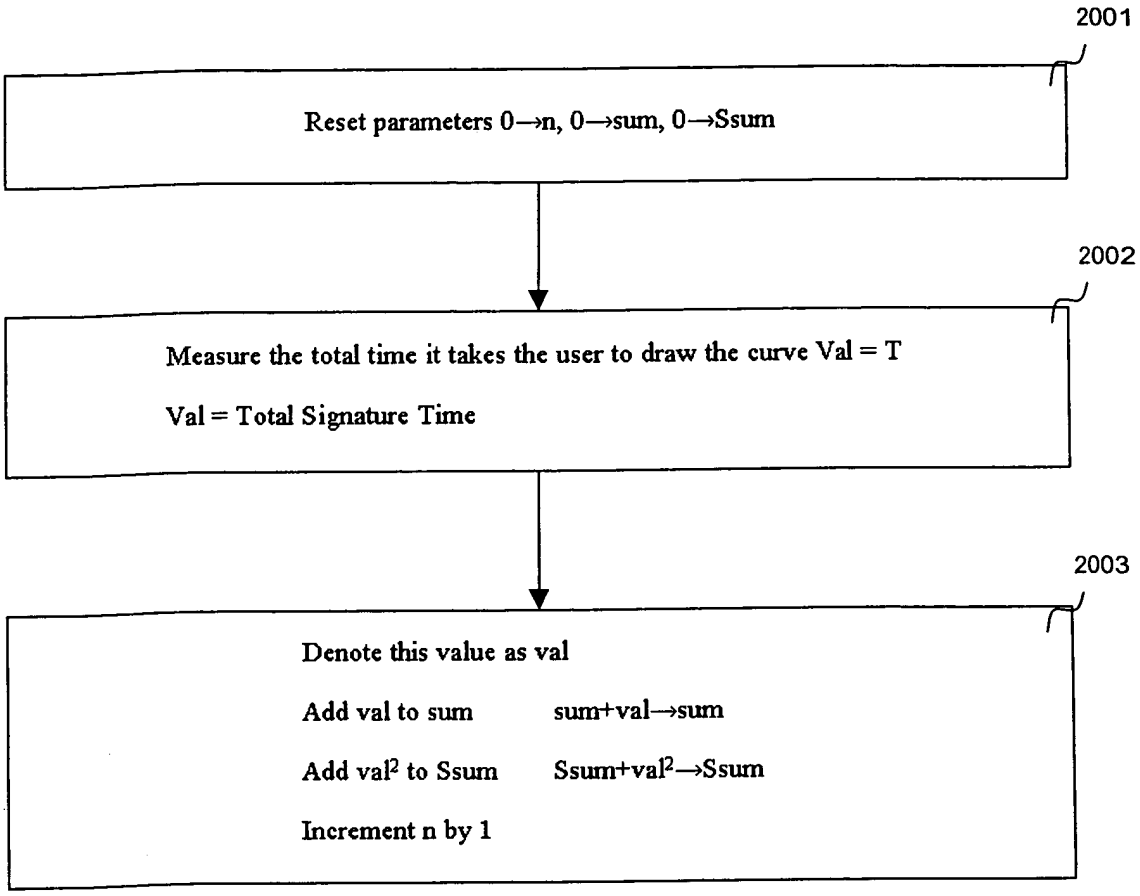

2001
Reset parameters 0→n, 0→sum, 0→Ssum
2002
Measure the total time it takes the user to draw the curve Val = T
Val = Total Signature Time
2003
Denote this value as val
Add val to sum
sum+val→sum
Add val² to Ssum
Ssum+val²→Ssum
Increment n by 1

Figure 20

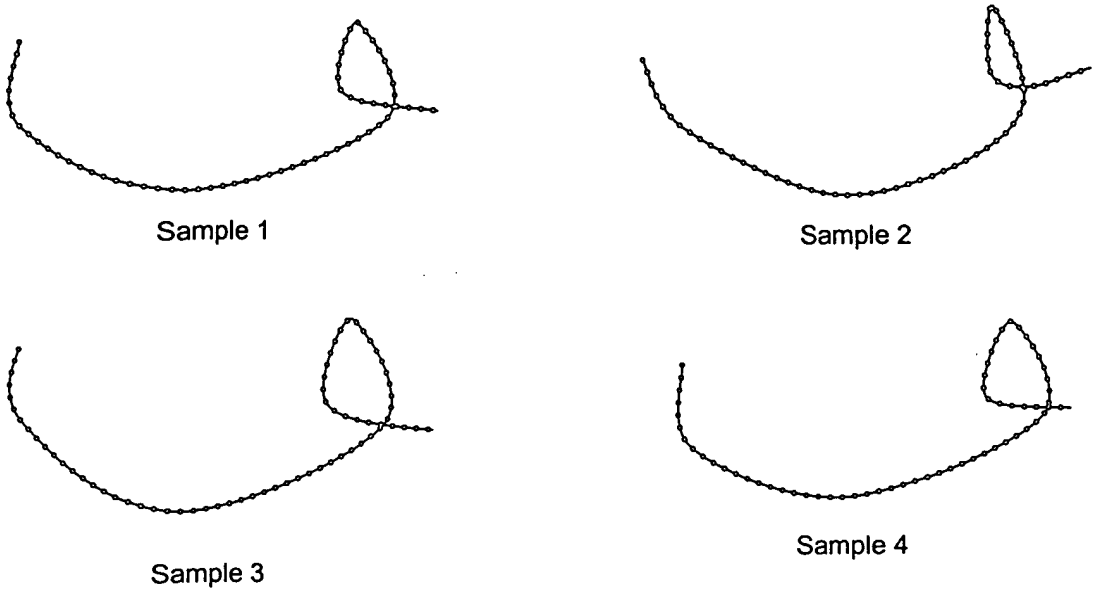
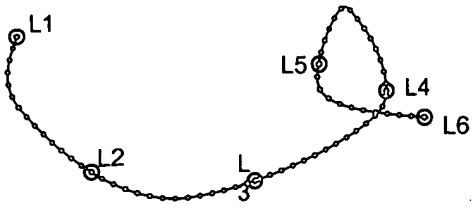
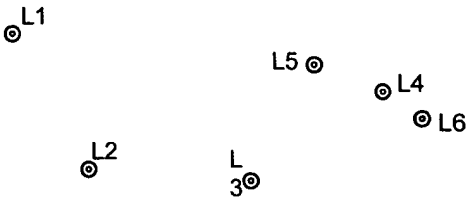


Figure 22



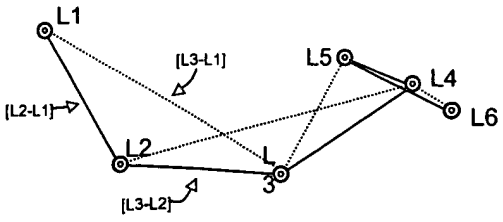
Sample 3 with Agents or Nodes
Located Every n (10) Samples

Figure 23



Agents or Nodes
From Sample 3

Figure 24

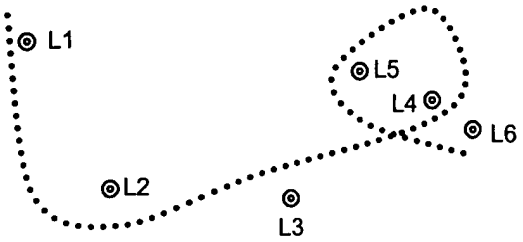


Form Geometric Characteristic Vector:

$$Val_i = \frac{[L_{i+1} - L_i] - [L_i - L_{i-1}]}{\text{ArcLength}\,[L_{i+1} \rightarrow L_{i-1}]}$$

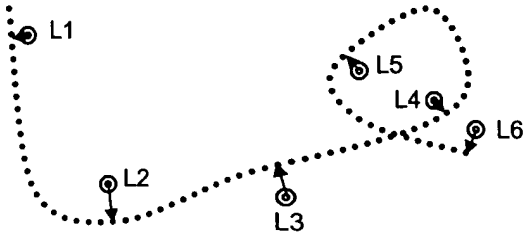
Store n-2 Geometric Characteristic Vectors in User
Profile Database as a User Authenticated Signature Profile

Figure 25



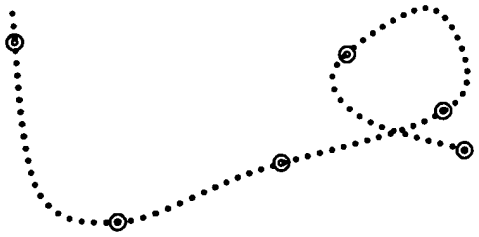
Sampled Signature with
Nodes or Agents As Stored

Figure 26



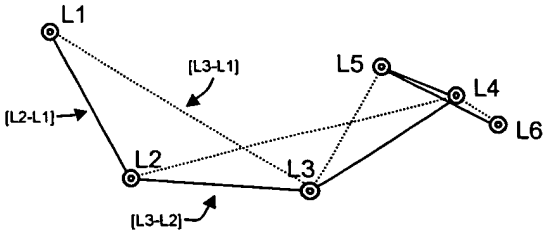
FCM to Migrate Nodes Toward Curve

Figure 27



Nodes Located on Curve by FCM

Figure 28



Form Geometric Characteristic Vector:

$$Val_i = \frac{[L_{i+1} - L_i] - [L_i - L_{i-1}]}{\text{ArcLength}\,[L_{i+1} \rightarrow L_{i-1}]}$$

Store n-2 Geometric Characteristic Vectors in User
Profile Database as a User Authenticated Signature Profile

Figure 29

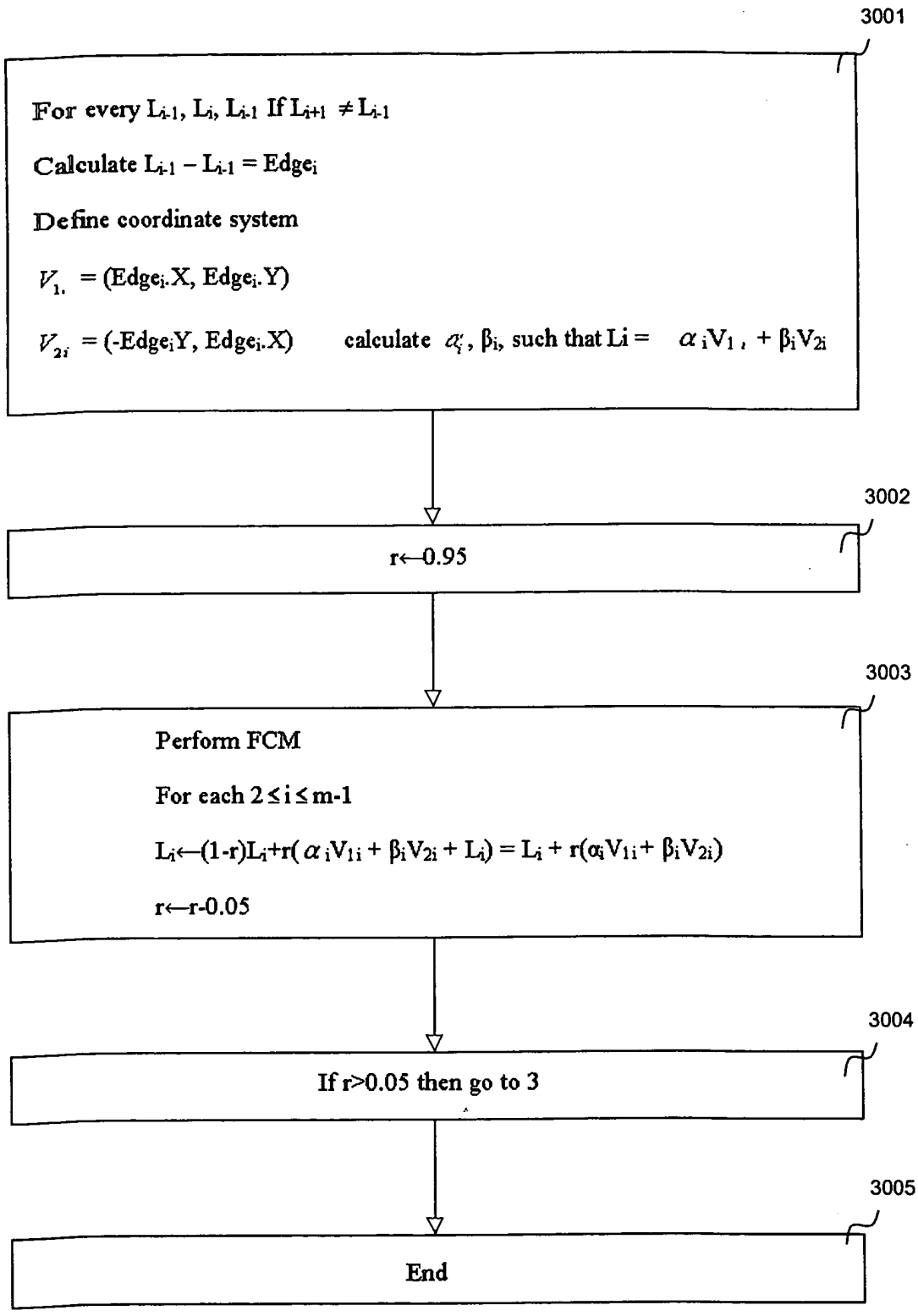

3001
For every L_{i-1}, L_i, L_{i-1} If $L_{i+1} \neq L_{i-1}$
Calculate $L_{i-1} - L_{i-1} = Edge_i$
Define coordinate system
$V_{1.}$ = (Edge$_i$.X, Edge$_i$.Y)
V_{2i} = (-Edge$_i$Y, Edge$_i$.X) calculate α_i, β_i, such that Li = $\alpha_i V_{1,} + \beta_i V_{2i}$
3002
r←0.95
3003
Perform FCM
For each $2 \leq i \leq m-1$
$L_i \leftarrow (1-r)L_i + r(\alpha_i V_{1i} + \beta_i V_{2i} + L_i) = L_i + r(\alpha_i V_{1i} + \beta_i V_{2i})$
r←r-0.05
3004
If r>0.05 then go to 3
3005
End

Figure 30